In [3]:
```
#Titanic - Exploratory Data Analysis (EDA)
#Dataset:train.csv
```

In [27]:
```
# imports
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# plotting defaults
%matplotlib inline
sns.set(style="whitegrid")
plt.rcParams['figure.figsize'] = (8,5)
```

In [28]:
```
df=pd.read_csv("train.csv")
df.head()
```

Out[28]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0

In [29]:
```
df.shape
df.info()
df.describe(include='all')
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
PassengerId    891 non-null int64
Survived       891 non-null int64
Pclass         891 non-null int64
Name           891 non-null object
Sex            891 non-null object
Age            714 non-null float64
SibSp          891 non-null int64
Parch          891 non-null int64
Ticket         891 non-null object
Fare           891 non-null float64
Cabin          204 non-null object
Embarked       889 non-null object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

Out[29]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
count	891.000000	891.000000	891.000000	891	891	714.000000	891.000000
unique	NaN	NaN	NaN	891	2	NaN	NaN
top	NaN	NaN	NaN	Birkeland, Mr. Hans Martin Monsen	male	NaN	NaN
freq	NaN	NaN	NaN	1	577	NaN	NaN
mean	446.000000	0.383838	2.308642	NaN	NaN	29.699118	0.523008
std	257.353842	0.486592	0.836071	NaN	NaN	14.526497	1.102743
min	1.000000	0.000000	1.000000	NaN	NaN	0.420000	0.000000
25%	223.500000	0.000000	2.000000	NaN	NaN	20.125000	0.000000
50%	446.000000	0.000000	3.000000	NaN	NaN	28.000000	0.000000
75%	668.500000	1.000000	3.000000	NaN	NaN	38.000000	1.000000
max	891.000000	1.000000	3.000000	NaN	NaN	80.000000	8.000000

In [30]:
```
df.isnull().sum().sort_values(ascending=False)
```

Out[30]:
```
Cabin          687
Age            177
Embarked         2
Fare             0
Ticket           0
Parch            0
SibSp            0
Sex              0
Name             0
Pclass           0
Survived         0
PassengerId      0
dtype: int64
```

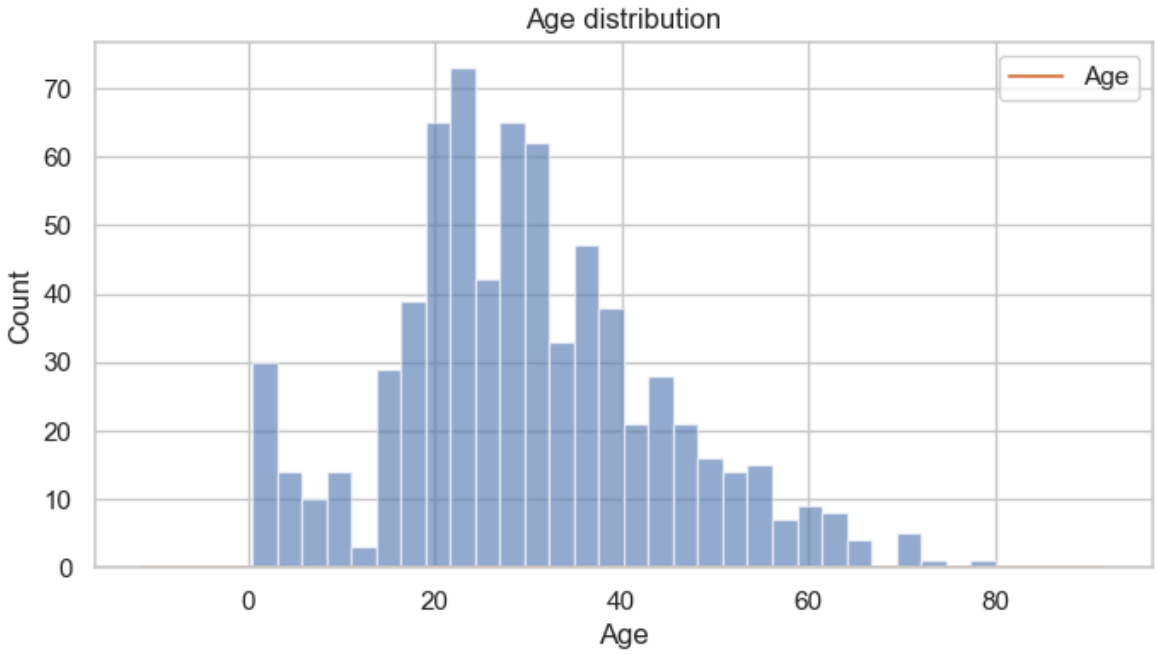
The cloumns which are having missing values are - Age,Cabin,Embarked

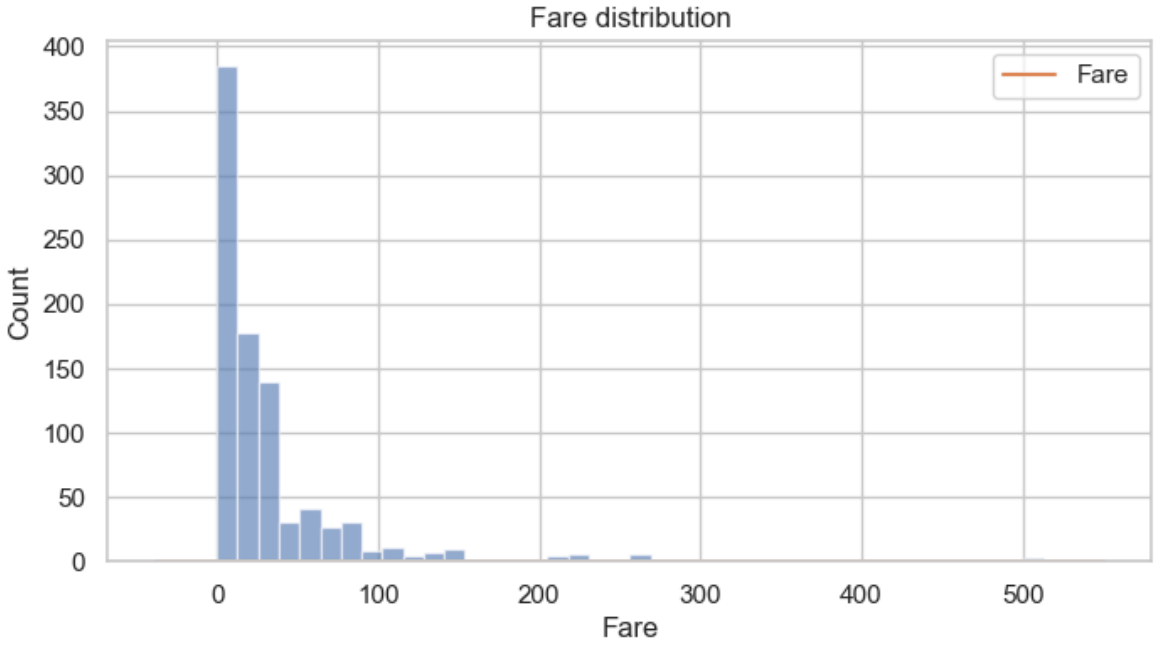
In [33]:
```
!pip install --upgrade seaborn
```

```
Requirement already satisfied: seaborn in c:\users\adnan\appdata\local\programs\p
ython\python37\lib\site-packages (0.10.1)
Collecting seaborn
  Downloading seaborn-0.12.2-py3-none-any.whl.metadata (5.4 kB)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\adnan\appdata\loc
al\programs\python\python37\lib\site-packages (from seaborn) (1.19.2)
Requirement already satisfied: pandas>=0.25 in c:\users\adnan\appdata\local\progr
ams\python\python37\lib\site-packages (from seaborn) (0.25.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\adnan\appdata
\local\programs\python\python37\lib\site-packages (from seaborn) (3.1.1)
Requirement already satisfied: typing_extensions in c:\users\adnan\appdata\local
\programs\python\python37\lib\site-packages (from seaborn) (4.7.1)
Requirement already satisfied: cycler>=0.10 in c:\users\adnan\appdata\local\progr
ams\python\python37\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.
11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\adnan\appdata\local
\programs\python\python37\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seabor
n) (1.4.5)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\use
rs\adnan\appdata\local\programs\python\python37\lib\site-packages (from matplotli
b!=3.6.1,>=3.1->seaborn) (3.1.4)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\adnan\appdata\loc
al\programs\python\python37\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seab
orn) (2.9.0.post0)
Requirement already satisfied: pytz>=2017.2 in c:\users\adnan\appdata\local\progr
ams\python\python37\lib\site-packages (from pandas>=0.25->seaborn) (2024.2)
Requirement already satisfied: six>=1.5 in c:\users\adnan\appdata\local\programs
\python\python37\lib\site-packages (from python-dateutil>=2.1->matplotlib!=3.6.1,
>=3.1->seaborn) (1.17.0)
Downloading seaborn-0.12.2-py3-none-any.whl (293 kB)
   ---------------------------------------- 293.3/293.3 kB 2.6 MB/s eta 0:00:00
Installing collected packages: seaborn
  Attempting uninstall: seaborn
    Found existing installation: seaborn 0.10.1
    Uninstalling seaborn-0.10.1:
      Successfully uninstalled seaborn-0.10.1
Successfully installed seaborn-0.12.2
```

In [37]:
```
import matplotlib.pyplot as plt
import seaborn as sns

def hist_with_kde(series, bins=30, title=None):
    plt.close('all')
    data = series.dropna()
    plt.figure(figsize=(8,4))
    # Preferred: try seaborn.histplot (newer versions)
    try:
        sns.histplot(data, bins=bins, kde=True)
    except AttributeError:
        # Fallback for older seaborn
        plt.hist(data, bins=bins, alpha=0.6)
        try:
            sns.kdeplot(data)
        except Exception:
            pass
```

```
    if title:
        plt.title(title)
    plt.xlabel(series.name)
    plt.ylabel("Count")
    plt.show()


hist_with_kde(df['Age'], bins=30, title="Age distribution")
hist_with_kde(df['Fare'], bins=40, title="Fare distribution")
```

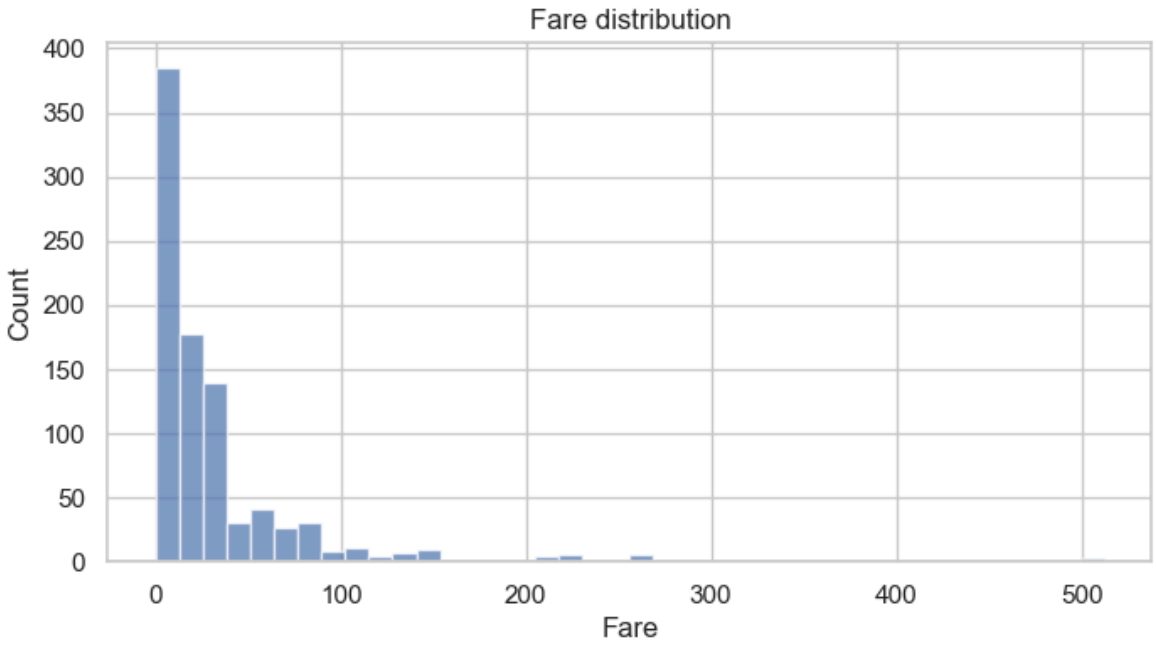


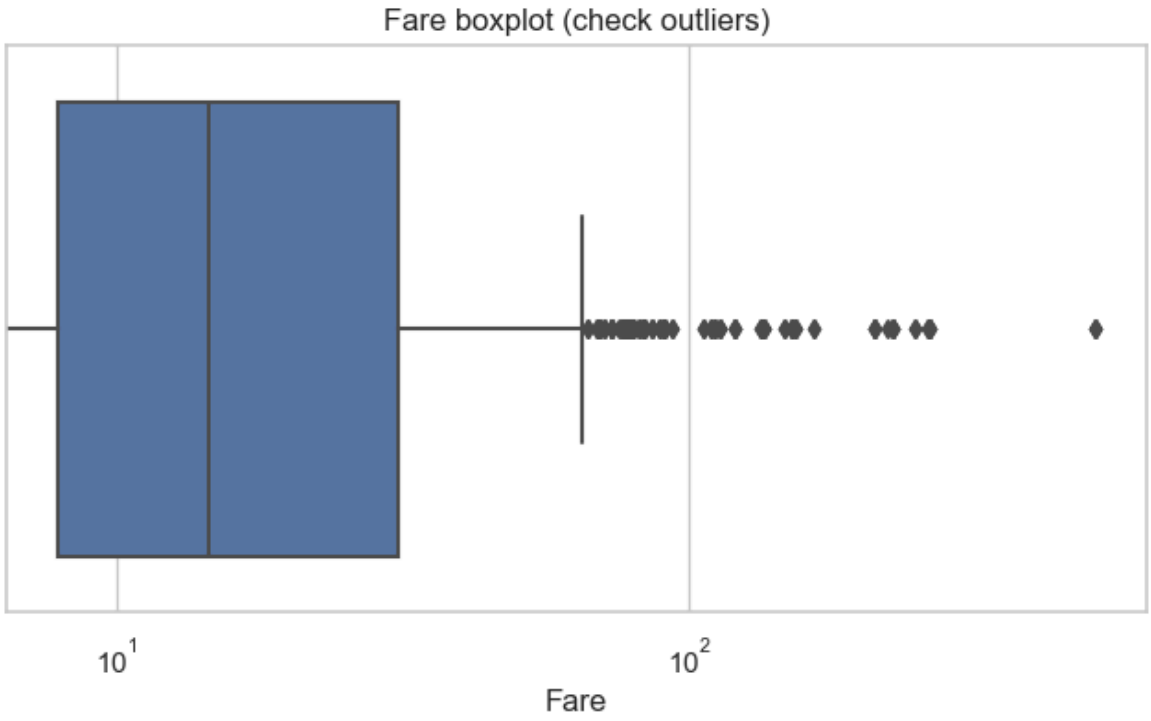


Observation-Most passengers are adults between 20–40. There are children under 12.

In [43]:
```
plt.figure(figsize=(8,4))
plt.hist(df['Fare'].dropna(), bins=40, alpha=0.7)
plt.title("Fare distribution")
plt.xlabel("Fare")
plt.ylabel("Count")
plt.show()
```

```
plt.figure(figsize=(8,4))
sns.boxplot(x=df['Fare'])
plt.title("Fare boxplot (check outliers)")
plt.xscale('log')
plt.show()
```

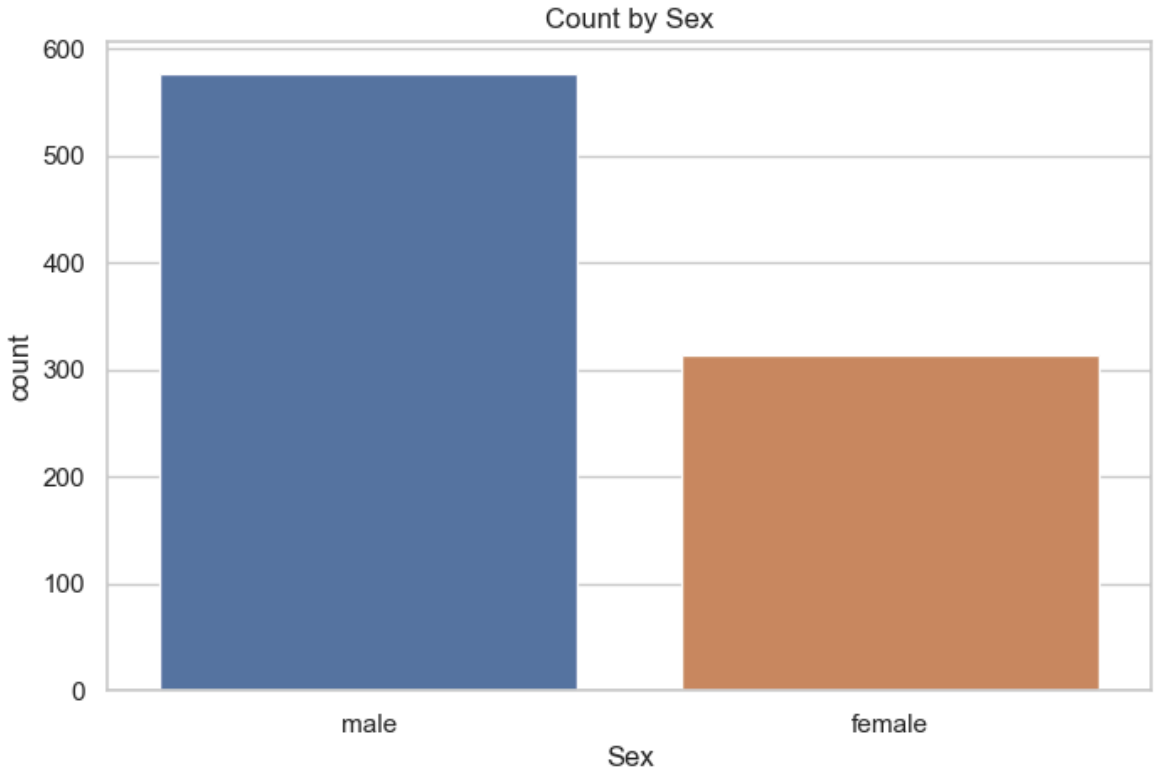


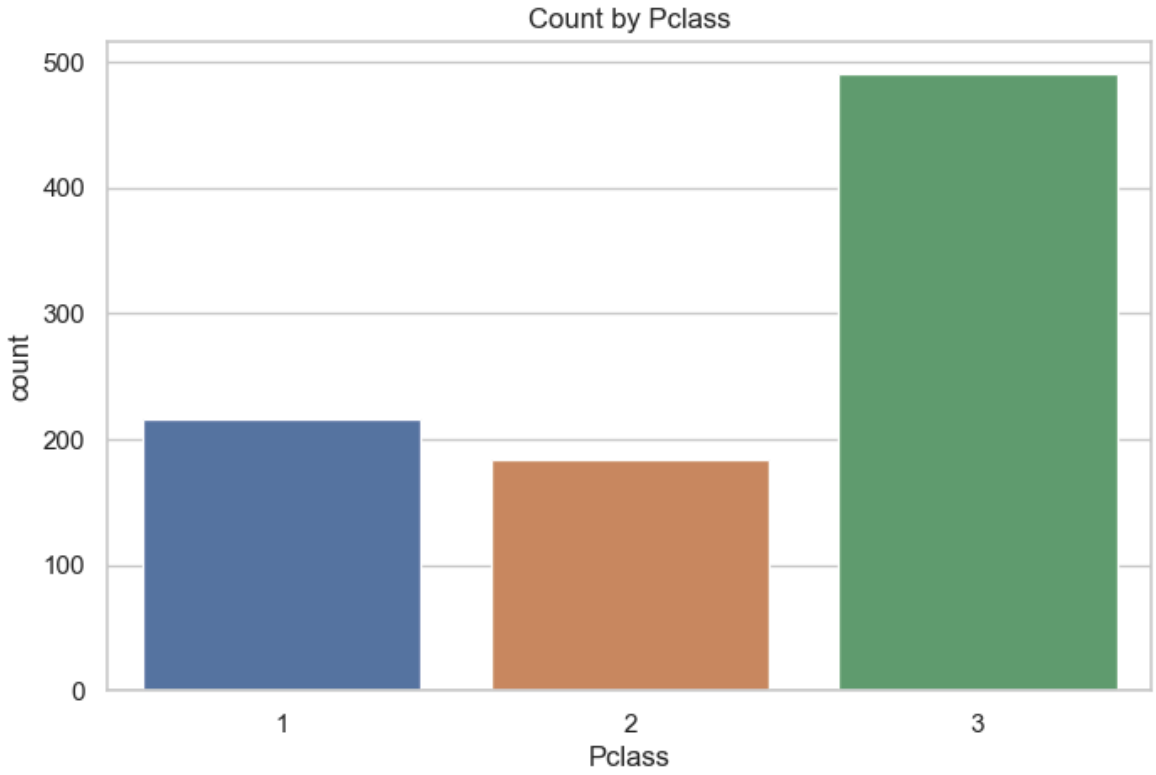


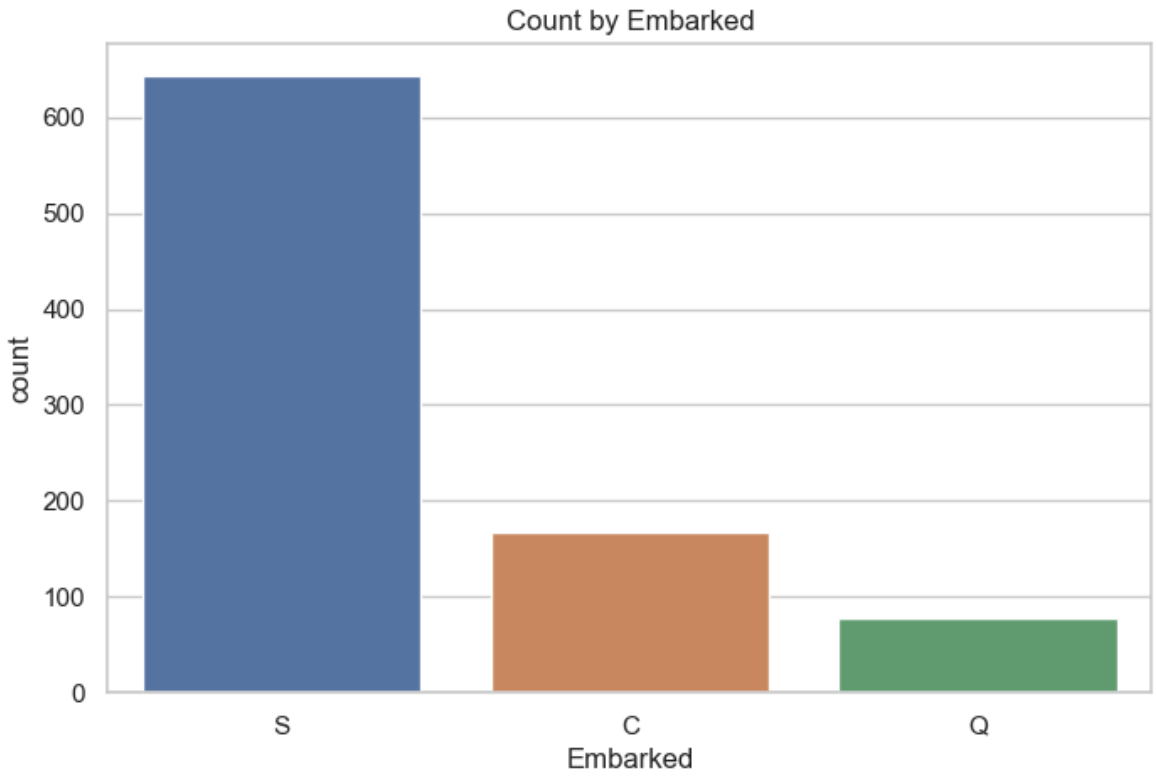
Observation-Fare is heavily right-skewed with large outliers (higher fares often in 1st class).

In [47]:
```
sns.countplot(data=df,x='Sex')
plt.title("Count by Sex")
plt.show()

sns.countplot(data=df, x='Pclass')
plt.title("Count by Pclass")
plt.show()
```

```
sns.countplot(data=df,x="Embarked")
plt.title("Count by Embarked")
plt.show()
```

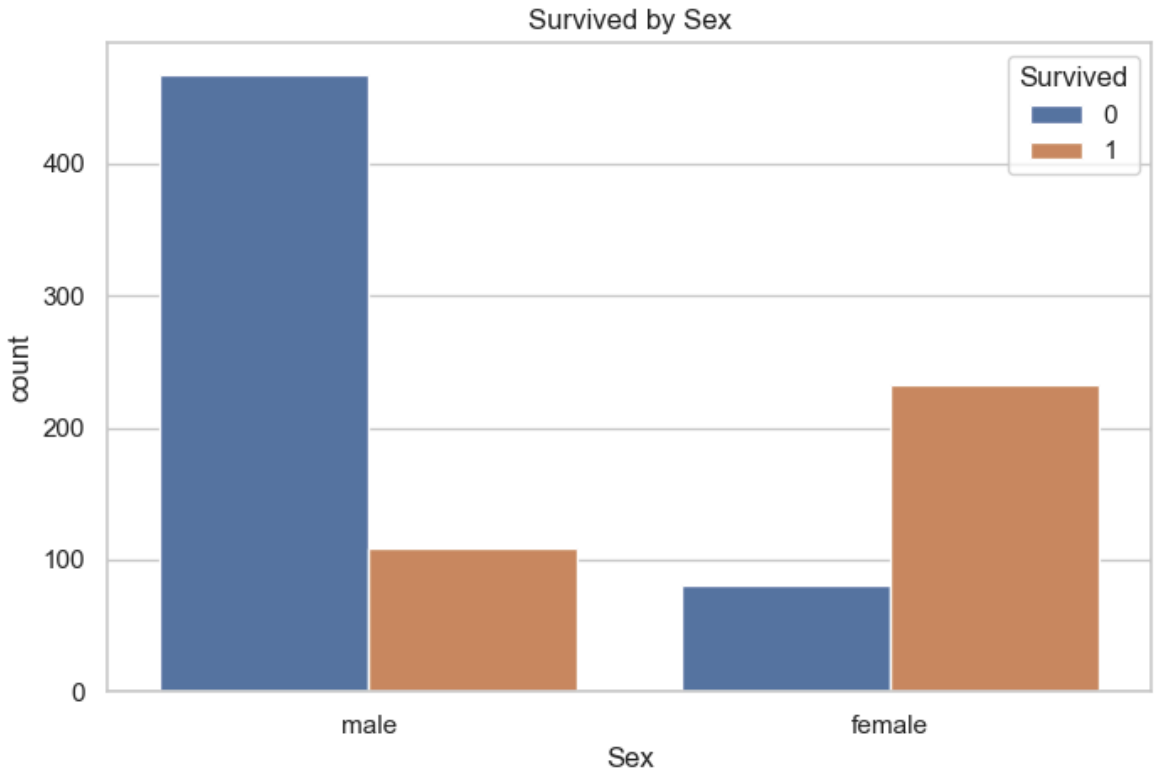






Observation-Most passengers embarked at S; majority are male.

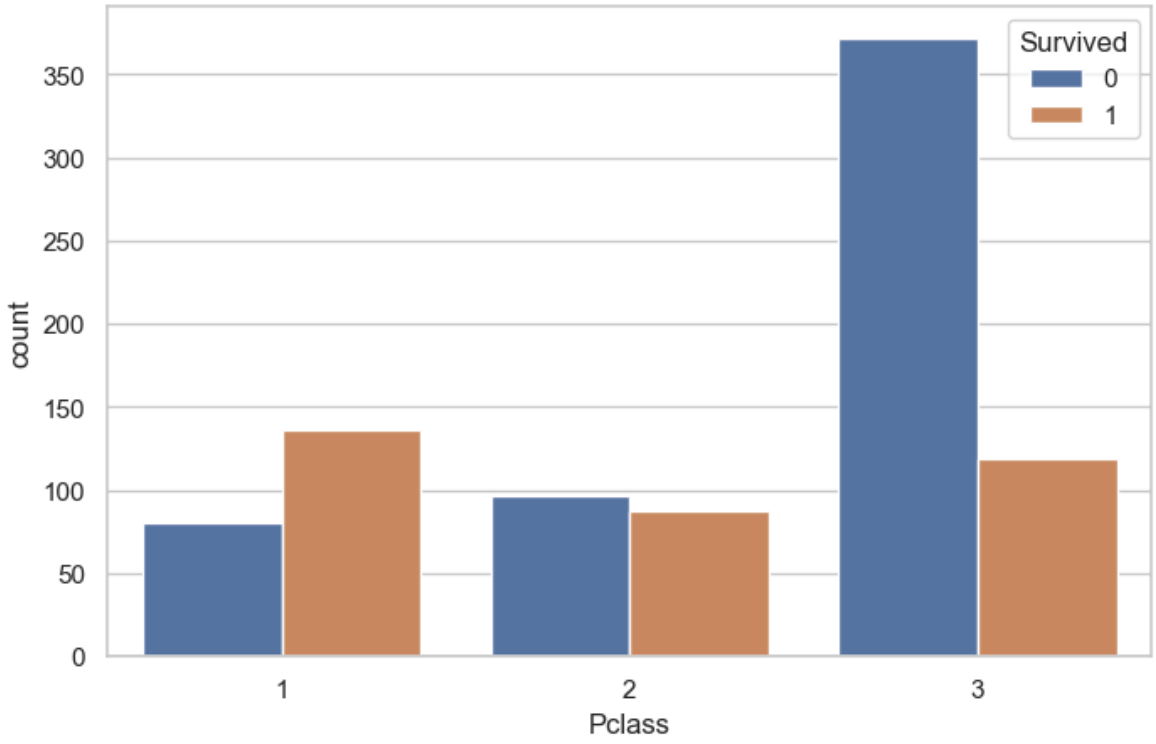
In [50]:
```
sns.countplot(data=df,x='Sex',hue= 'Survived')
plt.title("Survived by Sex")
plt.show()
df.groupby('Sex')['Survived'].mean().sort_values(ascending=False)
```



Out[50]:
```
Sex
female    0.742038
male      0.188908
Name: Survived, dtype: float64
```

Observation-Females have a much higher survival rate than males.

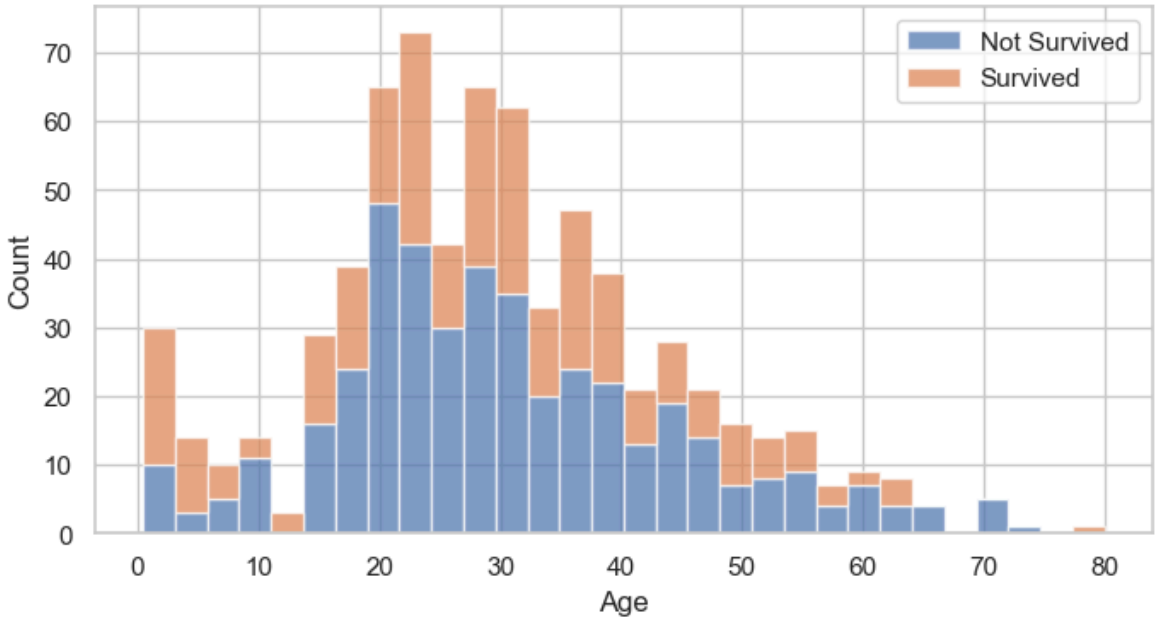
In [52]:
```
sns.countplot(data=df,x='Pclass',hue='Survived')
plt.title=("Survived By Pclass")
plt.show()

df.groupby('Pclass')['Survived'].mean()
```



Out[52]:
```
Pclass
1    0.629630
2    0.472826
3    0.242363
Name: Survived, dtype: float64
```

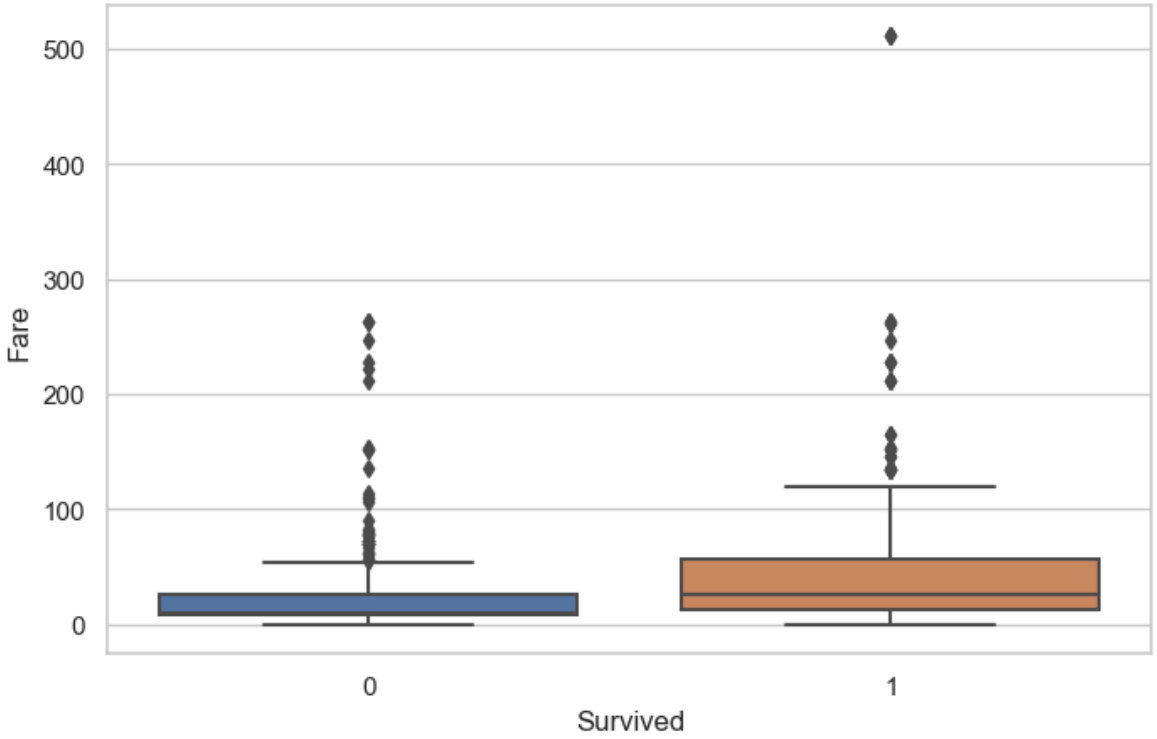
Observation-1st class shows higher survival proportion than 2nd and 3rd.

In [59]:
```
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="whitegrid")
age_not_survived = df.loc[df['Survived'] == 0, 'Age'].dropna()
age_survived = df.loc[df['Survived'] == 1, 'Age'].dropna()

plt.figure(figsize=(8,4))
plt.hist([age_not_survived, age_survived],
         bins=30,
         stacked=True,
         label=['Not Survived','Survived'],
         alpha=0.7)

plt.xlabel("Age")
plt.ylabel("Count")
plt.legend()
plt.show()
```



Observation-Children tend to have higher relative survival; check age bins.

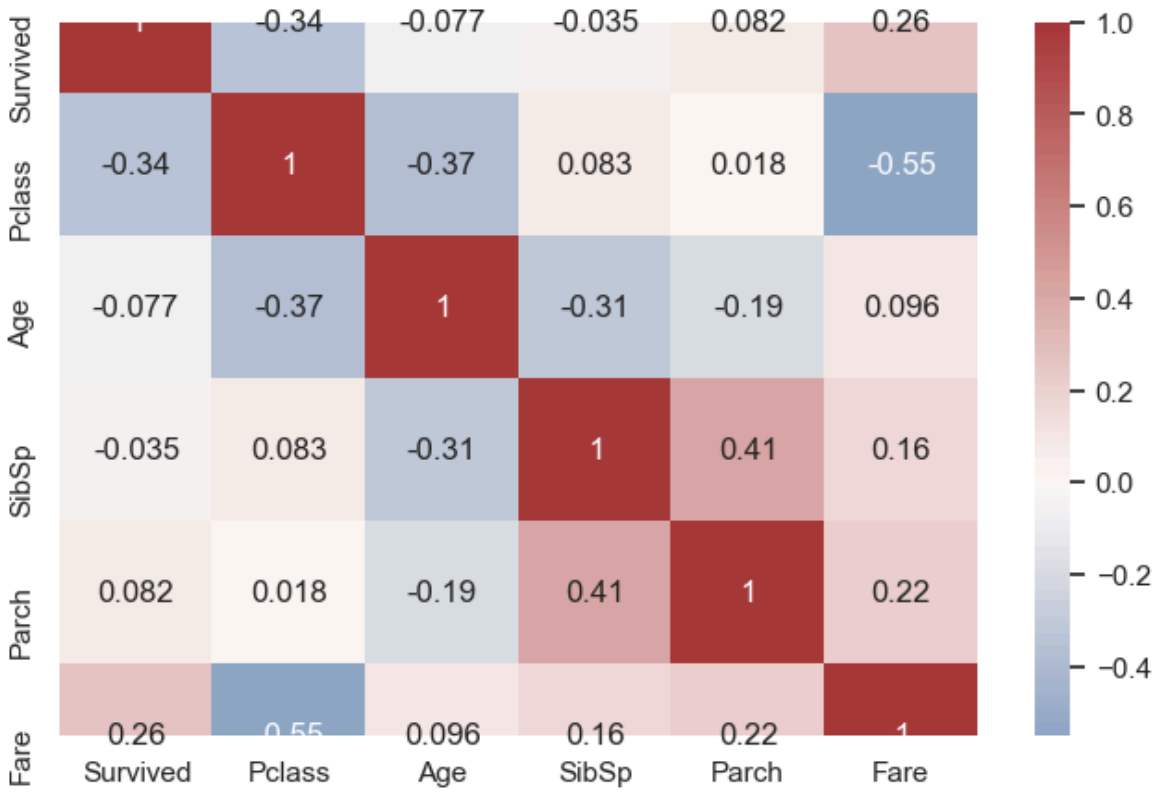
In [69]:
```
sns.boxplot(data=df,x='Survived',y='Fare')
plt.yscale=('log')
plt.show()
```



Observation-Survivors generally paid higher fares on average.

In [74]:
```
num_cols=['Survived','Pclass','Age','SibSp','Parch','Fare']
corr=df[num_cols].corr()
sns.heatmap(corr,annot=True,cmap='vlag',center=0)
print("Correlation matrix (numeric columns)")
plt.show()
```

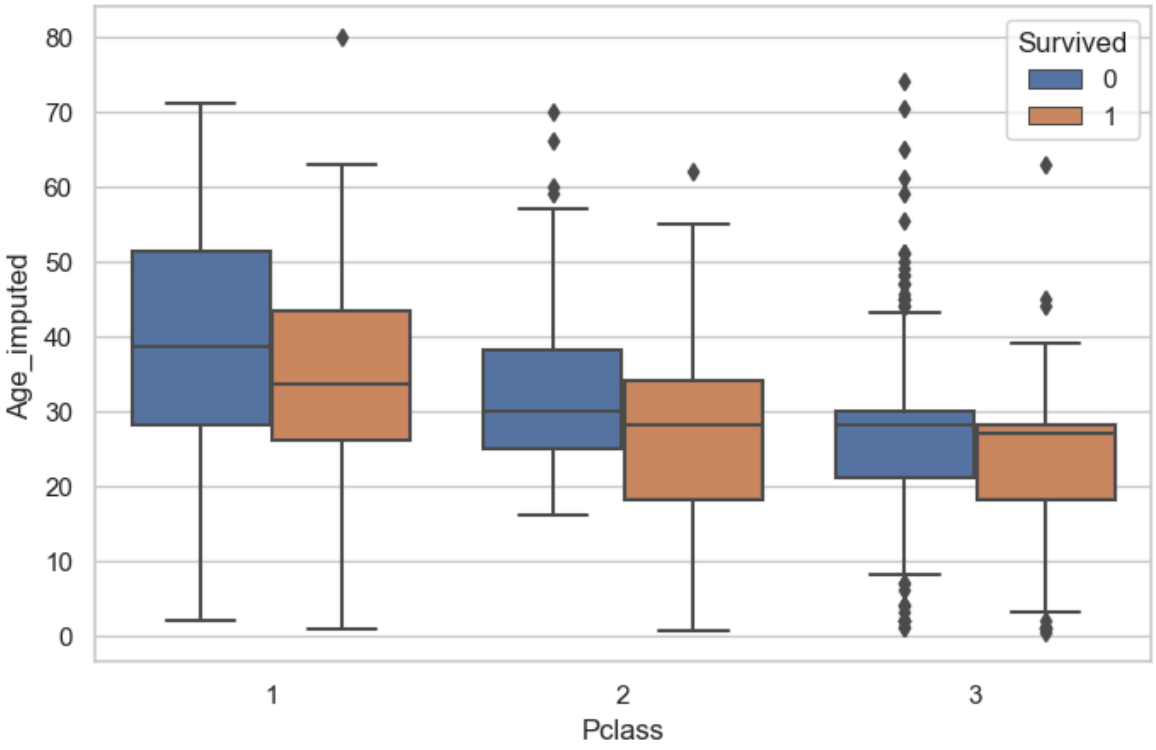
```
Correlation matrix (numeric columns)
```



Observation-Pclass and Fare have noticeable correlation with Survived; SibSp/Parch weaker.

In [80]:
```
# median imputation for quick analysis
df['Age_imputed'] = df['Age'].fillna(df['Age'].median())
sns.boxplot(data=df, x='Pclass', y='Age_imputed', hue='Survived')
print("Age (imputed median) by Pclass and Survival")
plt.show()
```

```
Age (imputed median) by Pclass and Survival
```



Observation-Median imputation is simple; prefer model-based or group-based imputation for production.

Summary of Findings

- Missing values: Age & Cabin (note: cabin mostly missing)
- Strong predictors: Sex (female), Pclass (1st), Fare (higher)
- Children show relatively higher survival — consider Age bins
- Titles and FamilySize are useful engineered features

Recommended next steps

1. Better Age imputation (group-based or predictive)
2. Build baseline model (Logistic Regression) + cross-validation
3. Report: include key plots above, short observations and methods

In [ ]: